Numerical Methods and Data Analysis with Spreadsheets

I. Objective

To use spreadsheets for studying the precision of several useful numerical methods (differentiation, integration, root solving) and to perform data analysis.

II. Introduction to Spreadsheets

One of our goals is to point out the strengths and weaknesses of different numerical techniques. Although programming in FORTRAN or C gives one complete control over how a calculation is done, there are many situations in which turning to a piece of commercial software makes more sense. We begin our introduction to spreadsheets by stating the type of work they do best:

- **Numerical algorithms defined by an iterative formula:** In many mathematical procedures, the value of the unknown depends on the value at a previous point. Examples include algorithms for *numerical differentiation, integration, root finding, and solving differential equations.* Formulas for several such methods are given in the Appendix.
- **Data analysis:** There are many ways to analyze experimental data using spreadsheets. Of great value is the ability to easily import, manipulate, modify, and graph data sets.

In this lab we will apply spreadsheets to several of these procedures using the spreadsheet **MS Excel**. In class, we will demonstrate several basic features of Excel that will assist you in completing today's exercises. These procedures are:

- Entering formulas and text.
- Copying, filling, and pasting cells.
- Defining relative and absolute cell addresses.
- Functions: Exp, Sin, Sum, Iseven, isodd. Macros and user-defined functions (optional).
- Creating graphs, style issues for publication in physics journals.

Today you will see how to use the capabilities of Excel to your advantage in numerical work and data analysis.

A note about graphs: Figures in scientific publications must follow established standards. For physics journals, the rules are explained in the *AIP Style Manual* (American Institute of Physics, New York, 4th edition, 1990). MS Excel **cannot** generate figures adequate for publication in AIP journals. For this purpose, we typically use **Sigmaplot** by **Jandel Scientific Software**. There may be other programs. Sigmaplot should be installed on at least one machine in B54. If you don't have anything better to do, try to make your graphs meet the requirements of the *AIP Style Manual*. Can you tell what's wrong with the figures generated by Excel (see *AIP Style Manual*)?

III. Exercises

A. Differentiation

Objective:	to compare derivative formulas of different orders
Where to begin:	start in MS Excel (with the template der1.xls, if you wish)
What to do:	create a spreadsheet that studies the first two derivatives of the function $f = \sin(x)$ **at x_0 = 1**
What to turn in to your instructor:	(1) graph(s) described below; (2) your comments on the results
What to put in log book:	the time you begin your work, problems, solutions, new commands, etc.

(1) **First derivative formulas:** We will compare the first derivative formulas given in the Appendix as a function of step size h at the point **x_0 = 1** for the function $f(x) = \sin(x)$.

Define the error $\Delta \equiv \left| f'_{exact} - f'_{numerical} \right|$. Create an appropriate spreadsheet that will allow you to study Δ as you change the step size h.

h	Δ for 3 pt formula	Δ for 5 pt formula
0.1		
0.01		
0.001		
0.0001		
0.00001		

(2) **Second derivative formulas:** Repeat the above exercise using the second derivative for both 3 point and 5 point formulas and now define $\Delta \equiv f''_{exact} - f''_{numerical}$.

Fill in the following table.

h	Δ for 3 pt formula	Δ for 5 pt formula
0.1		
0.01		
0.001		
0.0001		
0.00001		

(3) **Make log-log plot:** Use Excel's graphing functions to produce a log-log plot of Δ (on the y axis) versus h (on the x-axis) for both first and second derivatives comparing the 3 and 5 point formulas.

Comment on your results, especially why the error in the 5 point method increases below a certain step size. For a hint, get on Vincent and read **/home/physics/phys232/hint1**. You can also read this document on the Web page.

A detailed discussion of choosing the correct step size *h* can be found in *Numerical Recipes*, chapter 5.7, equation (5.7.5) and (5.7.8).

(4) **Save your work on your diskette:** Call your spreadsheet der1.xls.

B. Integration

Objective:	to compare the convergence of different integration formulas
Where to begin:	start in MS Excel
What to do:	create a spread sheet that performs the integral $\int_0^1 e^x dx$ using three methods
What to turn in to your instructor:	(1) graph described below; (2) your comments
What to put in log book:	the time you begin your work, problems, solutions, new commands, etc.

(1) **Compare three methods:** Create a spreadsheet for performing numerical integration that will compare the three methods given in the Appendix. You will calculate $\int_0^1 e^x dx$ for the Trapezoidal Rule, Simpson's Rule, and Bode's Rule using different numbers of intervals N. Define the error in the numerical calculation as $\Delta \equiv \left| \int_{exact} - \int_{numerical} \right|$ and use your spreadsheet to make the calculations necessary to fill in the table below.

N	$\Delta_{Trapezoidal}$	$\Delta_{Simpson}$	Δ_{Bode}
4			
8			
16			
32			
64			
128			

(2) **Make log-log plot:** Use Excel's graphing functions to produce a log-log plot of the calculation error Δ (on the y axis) versus N (on the x-axis) for each of the integration formulas.

Comment on the convergence of these different methods.

(3) **Save your work:** save your spreadsheet to your diskette calling it **int1.xls**.

C. Root Solving

Objective:	study the convergence of the Secant Method of finding roots
Where to begin:	start in MS Excel
What to do:	create a spread sheet that determines the zeros of $f(x) = x^2 - 5$
What to turn in to your instructor:	a copy of your spreadsheet
What to put in log book:	the time you begin your work, problems, solutions, new commands, etc.

(1) **Root finding:** create a spreadsheet that applies the Secant Method given in the

Appendix to solving for the roots of $f(x) = x^2 - 5$. Play around with the values of the first two points and see how quickly the method converges for very large initial values. Save your work on your disk in a file called **root1.xls**.

D. Data Analysis

Objective:	to study a nuclear magnetic resonance (NMR) spectrum for hydrogen using techniques explored above.
Where to begin:	FTP (to down-load), then go to MS Excel.
What to do:	down-load data **/home/physics/phys232/nmr** from Vincent and create a spreadsheet that performs the tasks below.
What to turn in to your instructor:	(1) a copy of your graphs (no data please); (2) an estimate of the ratio R (defined below).
What to put in log book:	the time you begin your work, problems, solutions, new commands, etc.

(1) **Down-load NMR data:** use FTP to down-load the the file /home/physics/phys232/nmr from the Physics 232 locker to your PC. Copy it to your A: drive and delete it from the C: drive. (You may also download the file from the PHY 232 homepage.)

In Excel, use the **File** menu to locate the **Open** command, and open the file **nmr** on your **A:** drive. Read the text given above the data (you may wish to delete the text after reading it).

(2) **Plot the data:** use Excel's graphing functions to graph the data. Notice there are two peaks, one big and one small.

(3) **Separate the peaks:** find a function f_{small} that will fit the small peak. We will assume that the small peak may be modeled by a Gaussian function of the form

$$f_{small}(x) = Ae^{-\frac{(x-B)^2}{2C^2}}$$

Determine the values of A, B, and C that give a reasonable fit to the small peak. We will now assume that

$$f_{data}(x) = f_{small}(x) + f_{big}(x)$$

Use this relation to determine $f_{big}(x)$

(4) **Integrate under the peaks:** use one of the integration methods in the Appendix to determine the areas under the small and large peaks. What is the ratio of areas R of the large to small peak?

(5) **Sum of squares (optional)**: How do you like the quality of your fit? Does your theory curve describe the data well? This is a typical example for a task that often appears in the analysis of experimental data: Assuming that a theory curve with N parameters (here: N=6) describes the data, find the parameter set that minimizes the deviation (sum of squares) between the theory and the data. What you have done above, is also known as **chi-by-eye**, and typically not very reliable, since it is subjective. You have determined the deviation (chi) by a visual inspection (eye). If you have time left, calculate the deviation (sum of squares) between theory and data and try to minimize it by making small changes to the parameters. Obviously, this procedure is very tedious. It would be nice to have a numerical algorithm, that would find the correct parameters for you. In most cases, the **Marquardt-Levenberg algorithm** (see *Numerical Recipes*) is most appropriate for this purpose. If there is time, we will address this algorithm in a later lab. By the way, do you think that a sum of two Gaussians adequately describes the data?

E. Access Maple

Before you leave, check to see if you can access **Maple** from your Vincent account. To do this, type % **add maple**, then % **maple** at the Vincent prompt. If it does not work, let us know.

IV. Appendix: Formulas for Numerical Algorithms

All of the differentiation and integration formulas are derived in the book *Computational Physics* by Steven Koonin which is on reserve in Parks Library. The formulas are obtained by writing the Taylor expansion of the function $f(x) = f_0 + xf' + \frac{x^2}{2!} f'' + \frac{x^3}{3!} f''' + \ldots$ where all the derivatives are evaluated $x_0 = 0$. See also Bronshtein and Semendyayev (Tables 7.10 and 7.13) or Abramowitz and Stegun (Chapter 25.4, Table 25.2).

A. Notation

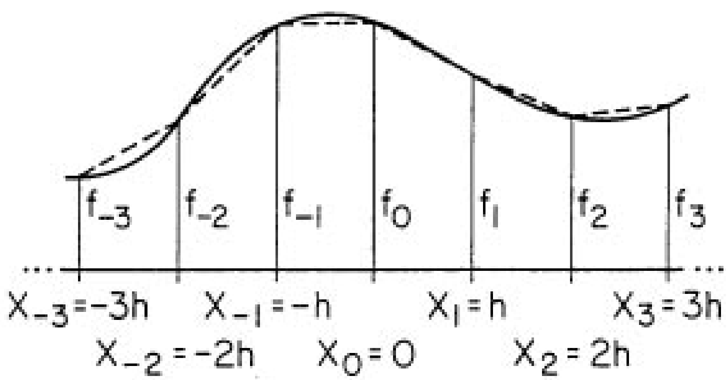
The following formulas are all based on the assumption that the function f(x) is defined on a number line of equally spaced coordinates. We define the following notation:

$$f_n \equiv f(x_0 + x_n) \text{ where } \mathrm{x_n} \equiv x_0 + nh \; (\mathrm{n} = 0, \pm 1, \pm 2, etc.)$$

The table below tells you how to think about this.

Think of ...	...as value of the function evaluated...
f_0	at the point of interest (x_0)
f_1, f_{-1}	one lattice space to the right / left of x_0
f_2, f_{-2}	two lattice spaces to the right / left of x_0

The figure below illustrates this.



B. Differentiation

First Derivative Formulas			
Method Name	**Formula**	**Error**	**Reference**
3 point formula	$f' = \frac{1}{2h}[f_1 - f_{-1}]$	O(h^2)	Koonin p.5
5 point formula	$f' = \frac{1}{12h}[f_{-2} - 8f_{-1} + 8f_1 - f_2]$	O(h^4)	Koonin p.5
Second Derivative Formulas			
Method Name	**Formula**	**Error**	**Reference**
3 point formula	$f'' = \frac{1}{h^2}[f_1 - 2f_0 + f_{-1}]$	O(h^2)	Koonin p.5
5 point formula	$f'' = \frac{1}{12h^2}[-f_{-2} + 16f_{-1} - 30f_0 + 16f_1 - f_2]$	O(h^4)	Koonin p.6

This method only works, if the function f whose derivative is to be calculated is known to arbitrary precision. When calculating derivatives of experimental data (which usually carry some noise), the resulting derivatives are even more noisy. Therefore, some smoothing needs to be done while calculating the derivatives. This can be achieved with the *Savitzky-Golay algorithm*, see the references on the Web site.

C. Integration

We will break up integrals over the range [a,b] into the smaller integrals (for Trapezoidal and Simpson's Rule)

$$\int_a^b f(x)dx = \int_a^{a+2h} f(x)dx + \int_{a+2h}^{a+4h} f(x)dx + \ldots + \int_{b-2h}^{b} f(x)dx$$

or (for Bode's Rule)

$$\int_a^b f(x)dx = \int_a^{a+4h} f(x)dx + \int_{a+4h}^{a+8h} f(x)dx + \ldots + \int_{b-4h}^{b} f(x)dx$$

and apply the formulas for the smaller integrals given below. Note that the interval length h is $h = \frac{b-a}{N}$ where N is the number of intervals. Note from the table that N must be chosen to be a multiple of 2 or 4 for the methods below. Higher-order formulas are given by Abramowitz and Stegun (but they are useless for numerical purposes).

Integration Formulas (Newton-Cotes, closed type)				
Method Name	**Formula**	**Error**	**N**	**Reference**
Trapezoidal Rule	$\int_{-h}^{h} f(x)dx = \frac{h}{2}\left(f_{-1} + 2f_0 + f_1\right)$	$O(h^3)$	multiple of 2	Koonin p.6
Simpson's Rule	$\int_{-h}^{h} f(x)dx = \frac{h}{3}[f_1 + 4f_0 + f_{-1}]$	$O(h^5)$	multiple of 2	Koonin p.6
Bode's Rule	$\int_0^{4h} f(x)dx = \frac{2h}{45}[7f_0 + 32f_1 + 12f_2 + 32f_3 + 7f_4]$	$O(h^7)$	multiple of 4	Koonin p.8

These formulas are only for small (infinitesimal) intervals. For large intervals, it is better to have *extended* formulas, see Eq. (4.1.11) on p. 127 of *Numerical Recipes* for an extended versions of the trapezoidal rule. These extended versions are easier to implement in a spreadsheet (without FOR loops). Instead of the extended version of Simpson's rule (4.1.13), it is easier to calculate the similar expression (4.1.14) using a spreadsheet, which is of the same order as Simpson's rule. Simpson's rule alternates between 3 and 4 (which is difficult to implement with a spreadsheet), whereas (4.1.14) can be impleted with a simple sum() statement.

These formulas are called *closed*, since they require evaluating the function at both end points and in between. There are also *open* and *semi-open* formulas, which can be convenient, if the integral diverges at the end.

These numerical methods can also be used to integrate experimental data, which are made up of a signal and noise. Since the integral is a linear functional, the noise is supposed to average out. The same is not true for calculating numerical derivatives, see above.

D. Root Solving

Method Name	Recursion Formula	Reference
Secant Method	$x_{i+1} = x_i - f(x_i)\left(\frac{x_i - x_{i-1}}{f(x_i) - f(x_{i-1})}\right)$	Koonin p.12

V. Excel Macros

You probably found that calculating the integrals is comlicated without having a FOR loop in your spreadsheet. The only way I could do it (without loops) was to create a large spreadsheet to store my intermediate values and then add the intermediate values with the SUM function.

Actually, you can write your own programs (also called MACROs) in MS Excel. This is a sample MACRO, which calculates the integral for a function, whose values are passed as the first parameter. The step size is the second parameter. Note that the first parameter is an array, or a range of cells, containing all the function values.

```
Function Simpson(values, step)
   N = values.Count
   ' number of points in the range
   Simpson = step * (values(1) + values(N)) / 3
   ' take the first and the last point
   i = 2
   Do While i < N
      Simpson = Simpson + 4 * step * values(i) / 3
      i = i + 2
   Loop
   i = 3
   Do While i < N
      Simpson = Simpson + 2 * step * values(i) / 3
      i = i + 2
   Loop
End Function
```

If you have time, try to write a similar macro to calculate the integral using Bode’s rule.